Evolutionary Intelligence (2025) 18:50
https://doi.org/10.1007/s12065-025-01034-w

RESEARCH PAPER

Elena: epigenetic learning through evolved neural adaptation

Kriuk Boris[1] · **Sulamanidze Keti**[2] · **Kriuk Fedor**[3]

Received: 15 January 2025 / Revised: 1 March 2025 / Accepted: 15 March 2025 / Published online: 12 April 2025
© The Author(s) 2025

Abstract
Despite the success of metaheuristic algorithms in solving complex network optimization problems, they often struggle with adaptation, especially in dynamic or high-dimensional search spaces. Traditional approaches can become stuck in local optima, leading to inefficient exploration and suboptimal solutions. Most of the widely accepted advanced algorithms do well either on highly complex or smaller search spaces due to the lack of adaptation. While recent approaches have attempted to address adaptability through various mechanisms, they typically require problem-specific tuning or sacrifice generalization capability for performance, failing to maintain consistent efficiency across different optimization tasks and varying problem scales. To address these limitations, we present ELENA (Epigenetic learning through evolved neural adaptation), a new evolutionary framework that incorporates epigenetic mechanisms to enhance the adaptability of the core evolutionary approach. ELENA leverages compressed representation of learning parameters improved dynamically through epigenetic tags that serve as adaptive memory. Three epigenetic tags (mutation resistance, crossover affinity, and stability score) assist with guiding solution space search, facilitating a more intelligent hypothesis landscape exploration. To assess the framework's performance, we conduct experiments on three critical network optimization problems: the Traveling salesman problem (TSP), the Vehicle routing problem (VRP), and the Maximum clique problem (MCP). Experiments indicate that ELENA achieves competitive results, often surpassing state-of-the-art methods on network optimization tasks.

Keywords Metaheuristic algorithms · Epigenetic learning · network optimization · Dynamic search spaces

1 Introduction

Optimization of complex networks is one of the fundamental challenges in computer science research. With the progression of computational resources availability, a great variety of conceptually different algorithms have been presented over the past decades to achieve competitive results in the domain of network optimization. Many approaches, such as Lin–Kernighan–Helsgaun heuristic [1], Genetic algorithm variations [2–4], Ant Colony Optimization [5], k-opt local search [6, 7] with sequential improvements have gained acknowledgment from both research community and industry across logistics, telecommunications, and biotechnology verticals.

The Traveling salesman problem (TSP) [8], first formalized by Karl Menger in 1930, remains a cornerstone problem that has driven network optimization algorithmic innovations for decades. The Vehicle routing problem (VRP) [9, 10], introduced by Dantzig and Ramser in 1959, extends TSP's complexity by incorporating multiple vehicles and capacity constraints, finding direct applications in logistics and delivery. The Maximum clique problem (MCP) [11], important for social network analysis, computational biochemistry and wireless network allocation, focuses on finding the largest complete subgraph within a network. Other flagman challenges include the Minimum spanning tree (MST) for network design, the Maximum flow problem (MFP) for resource allocation, and the Graph coloring

✉ Kriuk Boris
bkriuk@connect.ust.hk

Sulamanidze Keti
ksulamanidze.ieu2021@student.ie.edu

Kriuk Fedor
fedor.kriuk@student.uts.edu.au

[1] Department of Computer and Electronic Engineering, Hong Kong University of Science and Technology, New Territories, Hong Kong

[2] School of Science and Technology, IE University, Segovia, Spain

[3] Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, Australia

problem (GCP) for frequency assignment in signals and telecommunications. A diversified analysis of an algorithm's performance on several combinatorial problems helps to understand the generalization level and effectiveness of the approach.

Non-evolutionary approaches have demonstrated strong capabilities in network optimization tasks [12]. Popular methods leverage gradient-based learning, parallel programming, and advanced heuristics to navigate complex solution spaces. Deep neural networks, known for their ability to achieve high generalization levels [12–14], have shown promise in learning complex optimization patterns, achieving quality solutions for specific network instances [15, 16]. The integration of attention mechanisms and graph neural networks has further enhanced the ability of deep learning-based methods to capture structural relationships within networks [17], though they often require longer conversion time and careful hyperparameter tuning.

Evolutionary algorithms gained significant traction in network optimization due to the inherent ability to handle multi-objective optimization and maintain solution diversity [18, 19]. The population-based nature allows to implement parallel exploration of the solution space, making the algorithms particularly effective for problems with several local optima. The success of evolutionary approaches stems from their adaptability to different problem representations, ability to incorporate domain-specific knowledge through custom operators, and leniency to noisy or incomplete data. Additionally, the hybridization capabilities of evolutionary approaches with local search methods have a proven record of solving complex network optimization problems where traditional mathematical algorithms struggle to find global optima efficiently.

Despite these advantages, evolutionary approaches often struggle with maintaining solution validity and premature convergence, particularly in highly constrained problems like MCP [20]. Traditional genetic operators can produce invalid solutions, requiring expensive repair mechanisms or penalty functions.

Recent advances in biology, particularly in understanding epigenetics and horizontal gene transfer mechanisms, have inspired new approaches to evolutionary computation.

In this paper we introduce ELENA (Epigenetic Learning through Evolved Neural Adaptation)- a framework set to achieve significant advancement in the field of evolutionary approaches for network optimization. Our method extends best practices by introducing the concepts of compressed representation with epigenetic tags, adaptive mutation operators, and guided gene transfer mechanisms.

The algorithm's approach to parameter representation draws inspiration from the recognized work in deep learning, particularly the concepts of parameter compression and adaptive learning rates proposed by Hinton and Salakhutdinov in [21]. We believe that the integration of mentioned ideas with evolutionary optimization creates a unique hybrid approach that leverages the strengths of both fields. The compressed representation scheme used in ELENA reduces the search space dimensionality while maintaining solution quality, addressing a key challenge in large-scale optimization problems identified by Tian, Y., Zhang, X., Wang, C., and Jin, Y. in [22]. We conduct multiple experiments and show that ELENA is capable of achieving state-of-art results by being strong in both global and local solution space optimization.

2 Literature review

Early evolutionary approaches for network optimization primarily relied on genetic algorithms (GAs) and evolution strategies (ES). Holland's foundational work [23] established the theoretical framework for GAs, while Rechenberg and Schwefel's [24] developed first ES principles. Traditional genetic algorithms typically employ chromosome-like structures specific to network problems. For TSP, path and adjacency representation have emerged as popular encoding schemes [25]. While path representation maintains solution validity naturally, adjacency representation often requires repair mechanisms to ensure solution stability.

The evolution of these algorithms was significantly influenced by the concept of self-adaptation. Early research demonstrated that enabling algorithms to dynamically adjust their search parameters during execution, instead of relying on fixed operators, can lead to significantly improved results across various problem types [26]. While traditional evolutionary algorithms relied on fixed genetic operators, modern approaches have explored various forms of self-adaptation, where the algorithm's behavior evolves alongside the solutions. The evolutionary self-adaptation concept has proven particularly valuable in network optimization problems, where the complexity of solution spaces requires adaptive exploration strategies.

Later developments have introduced more complex evolutionary designs, including memetic algorithms. The integration of local search with evolutionary operations, introduced by Moscato [27], has shown superior performance in network optimization tasks. The algorithms combine global exploration through evolution with local exploitation through neighborhood search, especially effective for TSP and VRP variations. Differential Evolution (DE), proposed by Storn and Price [28], has demonstrated exceptional performance in continuous optimization problems. The adaptations for discrete networks have shown promising results, especially when combined with local search mechanisms.

Many researchers have earlier focused on developing efficient multi-objective evolutionary algorithms (MOEAs). Some advanced versions, such as NSGA-III [29] and MOEA/D [30], have shown solid results in handling large-scale optimization problems. These algorithms maintain diversity through reference points and decomposition strategies.

Later evolutionary approaches incorporated adaptive mechanisms, including parameter control with dynamic adjustment of mutation and crossover rates based on population diversity and fitness improvement [31], operator selection based on historical performance [32], and population size adaptation based on problem complexity and convergence patterns [33]. More recent research works have explored synergies between evolutionary algorithms and traditional machine learning, including the integration of surrogate models to predict fitness values, reducing computational cost [34], neural network hybridization [35], and reinforcement learning integration for adaptive operator selection and parameter tuning [36].

Among the most influential evolutionary algorithms that have shaped the field, several approaches stand out for their significant impact and widespread adoption. Genetic Algorithm with Adaptive Population Size (GAAPSo) [37], pioneered by Back and Schwefel, dynamically adjusts population size based on fitness improvements, showing notable adaptability across different problem domains. The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [38], developed by Hansen and Ostermeier, has become a standard in continuous optimization, known for theoretical foundations in self-adaptation mechanisms. Particle Swarm Optimization (PSO) [39] draws inspiration from social behavior in nature, demonstrating performance improvement in both continuous and discrete optimization tasks, and the Differential Evolution (DE) [40] has evolved into numerous variants, establishing itself as one of the most versatile evolutionary computation methods for both single and multi-objective optimization.

Still, several key challenges remain, including scalability barriers with large-scale networks [41], constraint handling while exploring the search space effectively [42], optimal parameter tuning [43], and developing efficient encoding schemes that maintain solution validity [44]. Even modern advanced methods leverage unique problem aspects, lacking generalized intuition behind implementation.

Emerging research directions explore quantum-inspired evolutionary algorithms for network optimization [45], integration of deep learning architectures with evolutionary search [46], application of transfer learning principles to evolutionary optimization [47], and development of distributed evolutionary algorithms for large-scale networks [48]. The field continues to evolve, leveraging insights from biology, machine learning, and theory of computation to address current limitations. The integration of attention mechanisms and transformer architectures with evolutionary computation also shows promise in achieving stronger performance [49]. Additionally, the development of compression techniques for genetic representations, inspired by advances in neural network pruning and quantization [50], offers potential solutions to the scalability challenges faced by evolutionary algorithms in large-scale network optimization problems. The trend towards self-adaptive and self-configuring algorithms continues to grow, and new approaches strive to apply meta-learning techniques to automatically adjust algorithm parameters based on problem characteristics [51].

Recent advances in gene transfer mechanisms have shown promising results in network optimization, particularly in workflow and job-shop scheduling domains. Notable works include Trans-Ptr-Nets-Based Transfer Optimization for multiobjective flexible job-shop scheduling in IIoT environments [52], scheduling constrained cloud workflow tasks via evolutionary multitasking optimization with adaptive knowledge transfer [53], knowledge transfer enabled diverse task scheduling for individualized requirements in industrial cloud platform [54], and evolutionary transfer optimization for flexible job-shop scheduling under uncertain processing times [55]. These approaches demonstrate the effectiveness of gene transfer in handling complex scheduling constraints and multiple objectives. ELENA builds upon such foundations by introducing a more generalized gene transfer mechanism that extends beyond scheduling-specific applications to broader network optimization problems.

3 Methodology

In this section we present ELENA (Epigenetic Learning through Evolved Neural Adaptation), and explain implementation details and differentiation aspects of the framework.

3.1 ELENA framework overview

The framework consists of several interconnected components that work together to explore the solution space efficiently.

The global exploration mechanism is implemented through the evolutionary process, which operates on multiple subpopulations simultaneously. We develop the mutation operator to support three types of modifications: swap, insert, and reverse mutations. The selection of mutation types and parameters to modify is performed dynamically. The crossover operation uses a custom order crossover that is controlled by systematic adaptations as well. Due to learning flexibility the resulting offspring inherits genetic material from both parents while maintaining solution validity. ELENA uses a dynamic

horizontal gene transfer mechanism which allows the exchange of stable solution segments between different subpopulations.

The local search is primarily handled by the 2-Opt method, which attempts to improve the solution by reversing subsequences of learning parameters. The method employs a first-improvement strategy, continuing until no better local modifications can be found. We leverage early stopping to terminate the evolution when no improvements are found, preventing unnecessary computational effort once the population has converged.

The algorithm maintains multiple subpopulations evolving in parallel during the evolution process. Parallel processing assists with accelerating the evaluation of individuals, and individual processing handles the application of mutation and local search operations for each solution independently. We strive to keep high genetic diversity and reduce the likelihood of non-optimal convergence, implementing the elitism strategy to preserve the best solutions while still exploring new ones through a combination of crossover and mutation operations.

The framework's effectiveness comes from its ability to balance exploration and exploitation through the interaction of local search, global search, and adaptation mechanisms described in the sects. 3.2, 3.3, 3.4 of this work.

3.2 Compressed representation with epigenetic tags

Traditional evolutionary algorithms often employ fixed mutation rates and predetermined crossover points which limits adaptation to local fitness landscapes. ELENA's epigenetic tags system introduces parameter-specific adaptation mechanisms that dynamically adjust evolutionary operations based on historical performance and current learning context, resulting in global and local intelligence.

We use a compressed representation of learning parameters, where each learning parameter (LP) is associated with three epigenetic tags. Epigenetic tags guide the learning process, dynamically controlling the core evolutionary algorithm's behavior. Compressed representation leverages a straightforward yet effective way to store parameters that need to be updated during population evolution. The schematic of compressed representation with epigenetic tags is summarized in Fig. 1. This approach allows the algorithm to store and retrieve parameters efficiently, reducing the number of compute operations.

The epigenetic tags for each LP include:

1. The mutation resistance, acting as a regularization mechanism and determining the likelihood of the parameter to undergo mutation during evolution. Parameters with high mutation resistance remain more stable and preserve inherent traits, while those with low resistance are more susceptible to change. Such an approach creates an adaptive mutation rate that varies across different parts of the solution space.

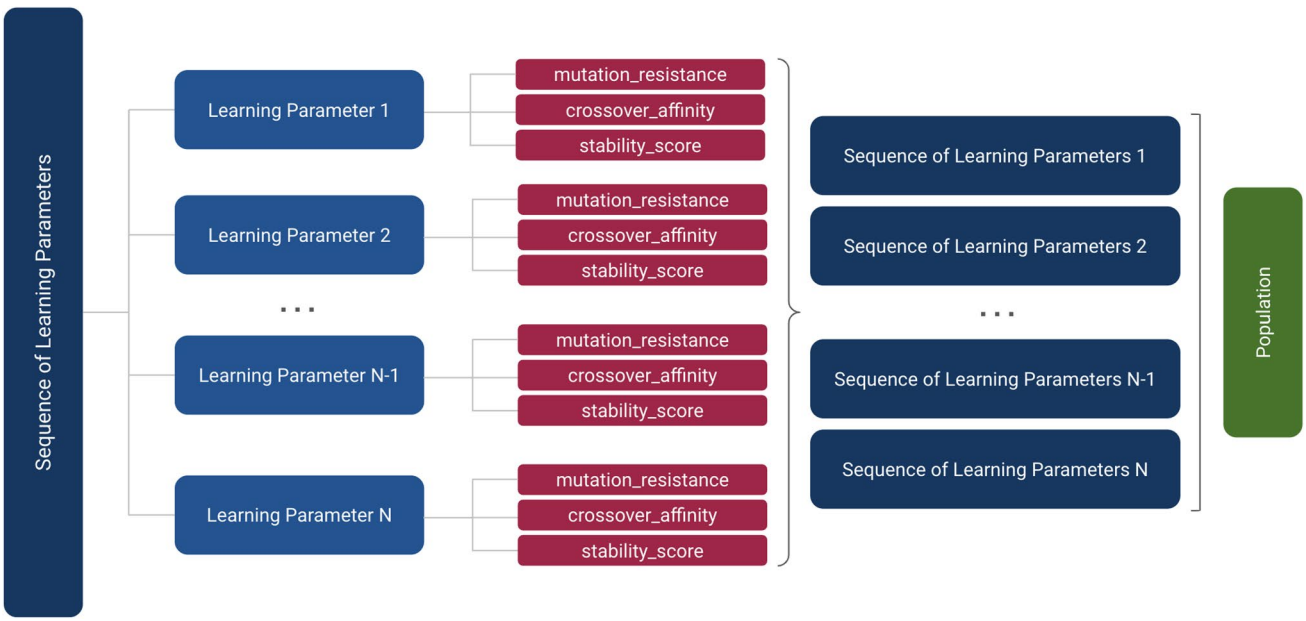

Fig. 1 Compressed Representation with Epigenetic Tags

2. The crossover affinity, identifying segments that are selected during genetic recombination. Parameters with high affinity are more likely to be kept together during crossover operations, preserving potentially beneficial parameter combinations. This idea helps to maintain blocks of high-performing solutions, still leaving space for genetic diversity.
3. The stability score, providing a historical performance metric for each parameter. The score increases when the parameter contributes to improvements and decreases otherwise. Such concept creates a feedback loop that influences both mutation and crossover operations, steering the evolution toward more promising regions of the search space.

Epigenetic tags responsible for mutation strength and performance stability are updated during 2-Opt improvement to normalize findings from local and global optimization. Mutation probability check occurs separately during mutation, followed by crossover selection. We show details of epigenetic tags updates in Algorithm 1.

We implement the fitness evaluation system through distance calculations with epigenetic tags to create a balanced approach between solution quality and performance stability (1). The strategy helps to prevent premature convergence while maintaining search efficiency.

$$F(LP) = 1/\sum d(LP_i, LP_{i+1}) \quad (1)$$

3.3 Hyperparameter choice for adaptive mutation and crossover operations

Adaptability is essential for an ML algorithm's ability to generalize well on complex tasks [56–58]. We explore the influence of the framework's starting environment on the convergence, testing various mutation rates and population sizes. We explore the combinations of population sizes equal to 15, 25, and 50 and initialization mutation rates of 0.2, 0.5, 0.7 respectively together with a crossover affinity of 0.7 on finding optimal solutions for TSP by generating random

Algorithm 1 Epigenetic Tag Updates During 2-Opt Improvement

for all tag T at time $t+1$ **do**
  if improvement > 0 **then**
    $mr(t+1) \leftarrow \min(1, mr(t) + \delta_{mr})$
    $ss(t+1) \leftarrow \min(1, ss(t) + \delta_{ss})$
  else
    $mr(t+1) \leftarrow \max(0, mr(t) - \delta_{mr})$
    $ss(t+1) \leftarrow \max(0, ss(t) - \delta_{ss})$
  end if
end for

where:
mr: mutation resistance
ss: stability score
δ_{mr}: mutation resistance step
δ_{ss}: stability score step

Table 1 Influence of ELENA's Starting Environment on Convergence

	ELENA								
	pop_size=15			pop_size=25			pop_size=50		
	mut=0.2	mut=0.5	mut=0.7	mut=0.2	mut=0.5	mut=0.7	mut=0.2	mut=0.5	mut=0.7
10 cities	2.903	2.903	2.903	2.903	2.903	2.903	2.903	2.903	2.903
25 cities	3.884	3.867	4.442	4.366	4.461	4.588	4.174	4.417	4.088
50 cities	5.857	6.090	5.586	5.934	5.933	5.856	5.887	5.814	5.715
100 cities	7.657	8.156	8.082	7.460	8.075	8.222	8.010	7.901	7.908
200 cities	10.850	11.023	11.014	11.180	10.881	11.184	10.930	11.014	10.612
Total	31.150	32.037	32.028	31.842	32.253	32.733	31.903	31.949	31.223

coordinate-sets for different numbers of cities. The results are summarized in Table 1.

We highlight the importance of optimal parameter initialization even within an adaptable frame. The superior performance seen with a higher initial mutation rate, particularly in larger solution spaces, suggests that higher initial mutation rates allow the system to establish effective mutation resistance patterns early in the evolutionary process. Stronger mutation rate helps to increase initial solution space exploration, making it a useful tool to improve performance of particular algorithm instances with smaller population sizes.

While common sense might suggest that larger populations would yield superior results through increased genetic diversity, our findings demonstrate that samples with moderate population sizes often achieve comparable or better performance. Such pattern is particularly evident in the mid-range problem sizes of 50 and 100 cities, where configurations with population sizes of 15 and 25 may outperform their larger counterparts. We associate this observation with the importance of finding a compromise between exploration and exploration to achieve strong performance on network optimization tasks.

3.4 Dynamic horizontal gene transfer

We integrate the horizontal gene transfer (HGT) mechanism to enable lateral genetic material exchange between subpopulations, mimicking bacterial conjugation processes. Unlike traditional HGT approaches that implement random or fitness-based transfer protocols, our implementation leverages epigenetic tagging system to dynamically control genetic material stability. ELENA's HGT operates through a dual-layer selection process where stable segments are identified based on the stability scores and transferred across population boundaries. The selection probability of 0.1 between any two individuals from different subpopulations maintains a balance between population diversity and convergence speed and has been empirically identified to be optimal.

We identify high-performing solution components using a stability threshold of 0.7 so that only well-tested genetic material is transferred. When stable segments are located, they are inserted into recipient individuals while maintaining solution validity by removing duplicate LPs before insertion. The approach differs from traditional evolutionary operators and allows to increase the speed of spreading beneficial traits across the entire population structure. The stability score threshold optimizes the trade-off between conservation of high-quality solution segments and maintenance of variation in the learning process. The controlled frequency of HGT operations, occurring every 5 generations, is specifically chosen to allow sufficient time for subpopulations to develop distinct characteristics while preventing premature convergence. Such temporal regulation combined with the subpopulation architecture helps to leverage the compute benefits of parallel programming in evolutions, and still optimize regularly the local search.

The mechanism's effectiveness is particularly evident in larger problem instances, where subpopulations tend to converge to different local optima. ELENA's design enables exchange of stable solution parts between subpopulations, combining locally optimal segments into globally superior solutions. The dynamic nature of the transfer process controlled by stability scores ensures that only proven beneficial traits are shared.

4 Experiments

We conduct a diversified series of experiments to evaluate the performance of the framework on fundamental network optimization problems. This section is divided into 4.1 Traveling salesman problem, 4.2 Vehicle routing problem, 4.3 Maximum clique problem where we present performance analysis for corresponding network optimization tasks.

4.1 Traveling salesman problem

The Travelling salesman problem (TSP) is a classic problem in combinatorial optimization, where the algorithm needs to find balance between local and global search to achieve strong performance on large instances. This section

Table 2 TSP Performance Comparison

	ELENA (Our)	Neirest neighbour	2-Opt for neirest neighbour	Simulated annealing	Christofides	Ant colony optimization	SOM	LKH-3	Concorde
10 cities	2.903	3.712	3.042	2.903	3.572	2.903	2.903	2.983	3.122
25 cities	3.884	5.159	4.073	4.843	5.527	4.250	4.153	4.460	4.186
50 cities	5.857	7.379	5.892	8.199	6.372	6.066	5.725	5.681	6.948
100 cities	7.657	9.895	8.529	16.701	8.364	8.224	7.689	8.016	10.064
200 cities	10.850	12.528	12.158	36.503	13.208	13.162	11.847	10.943	13.034
Total	31.150	38.674	33.694	69.150	37.043	34.605	32.314	32.083	37.405

provides an in-depth performance analysis of various algorithms designed to solve TSP, as presented in Table 2. The algorithms compared include ELENA with population size of 15 and starting mutation of 0.2, Nearest neighbor, Nearest neighbor with 2-Opt, Simulated annealing, Christofides algorithm, Ant colony optimization (ACO), Self-organizing map (SOM), Lin–Kernighan Heuristic (LKH-3), and Concorde. The approaches are evaluated based on the solution quality (tour length) for different numbers of cities. We generate a set of data points with random coordinates to diversify the comparison setup.

It can be observed that ELENA demonstrates competitive performance across all scales. Unlike most fundamental approaches that struggle with either scalability or local optimization, ELENA manages to achieve solution consistency due to its adaptable nature.

For smaller hypothesis spaces ELENA identifies optimal routes due to the combination of dynamic learning and the 2-Opt local improvement strategy. The 2-Opt method works particularly well in this context, as the limited number of connections means that reversing segments can lead to significant improvements. The idea to perform local normalization dynamically and not after each evolution cycle boost the operational efficiency even for smaller instances without affecting the general effectiveness of the approach.

ELENA's design is inherently scalable, making it well-suited for larger city counts. The structure allows to handle increased complexity without a corresponding increase in computational burden. The use of subpopulations helps maintain diversity and exploration across a larger solution space. As the problem size increases, the algorithm can dynamically adjust configurations to better navigate the more complex solution landscape.

The 2-Opt strategy remains effective even with larger instances. While the number of potential routes grows exponentially, our approach can iteratively refine solutions to find improvements that might not be straightforward. The ongoing refinement is crucial in large-scale problems as local optimization can lead to significant gains.

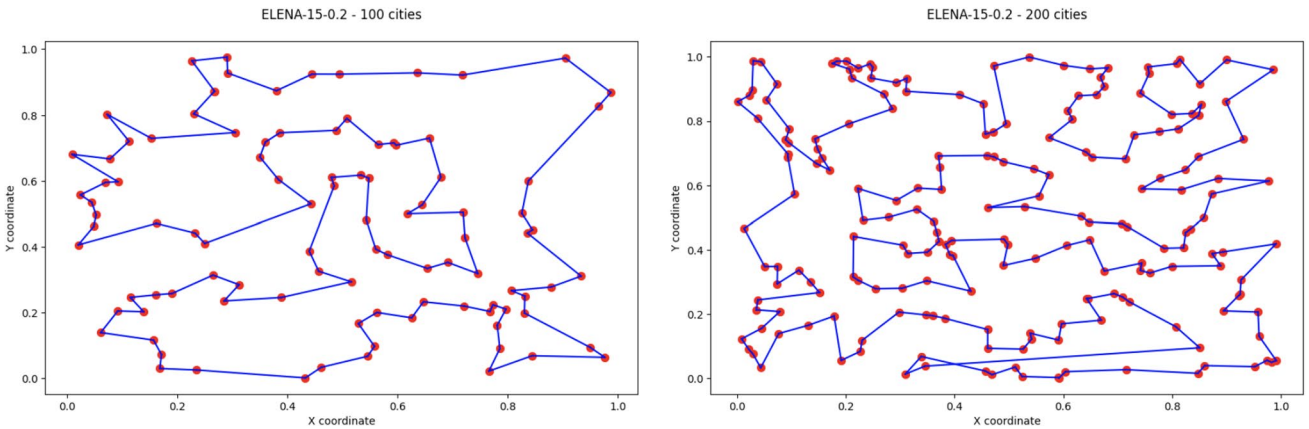

Fig. 2 ELENA-15−0.2 TSP Optimal Paths Comparison

Table 3 VRP Performance Comparison

	ELENA (Our)	Simulated annealing	Neirest neighbour	Ant cColony optimization	LKH-3
10 cities, vehicles 2, capacity 30	3.986	3.986	4.408	4.010	4.408
25 cities, vehicles 3, capacity 50	6.935	6.688	8.577	7.932	8.152
50 cities, vehicles 5, capacity 100	7.312	7.638	8.669	8.216	8.437
100 cities, vehicles 7, capacity 200	10.033	10.135	11.253	10.961	10.844
200 cities, vehicles 10, capacity 400	12.914	14.241	14.241	14.548	12.979
Total	41.180	42.685	47.108	45.667	44.818

We also reveal examples of optimal paths found by ELENA in Fig. 2.

4.2 Vehicle routing problem

Vehicle Routing Problem (VRP) introduces new complications to TSP through controlling a number of additional parameters affecting the solution space size. Due to increased task complexity, we initialize ELENA with larger and more lenient network parameters (population size of 50 and starting mutation rate of 0.5) and compare its performance against common algorithms. We create a small dataset by assigning random cities coordinates with manually predefined parameters for initial testing, and results are summarized in Table 3.

The conclusions made in 4.1 Traveling Salesman Problem are further reinforced by new observations. ELENA exhibits consistent performance across all solution space sizes which can be attributed to its adaptable nature. Adaptability is essential, as it enables ELENA to make real-time adjustments based on the evolving landscapes which can differ dramatically in VRP.

We state that evolutionary approaches can significantly outperform most traditional algorithms on network optimization complex instances since the sheer number of potential routes easily overwhelms static algorithms. ELENA's

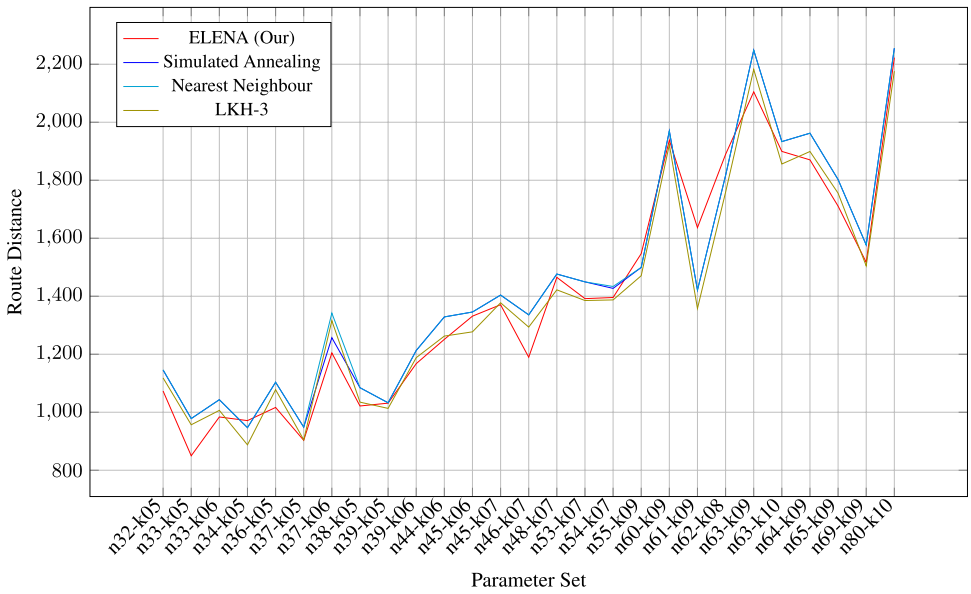

Fig. 3 VRP Augerat-1995 set A Performance Comparison

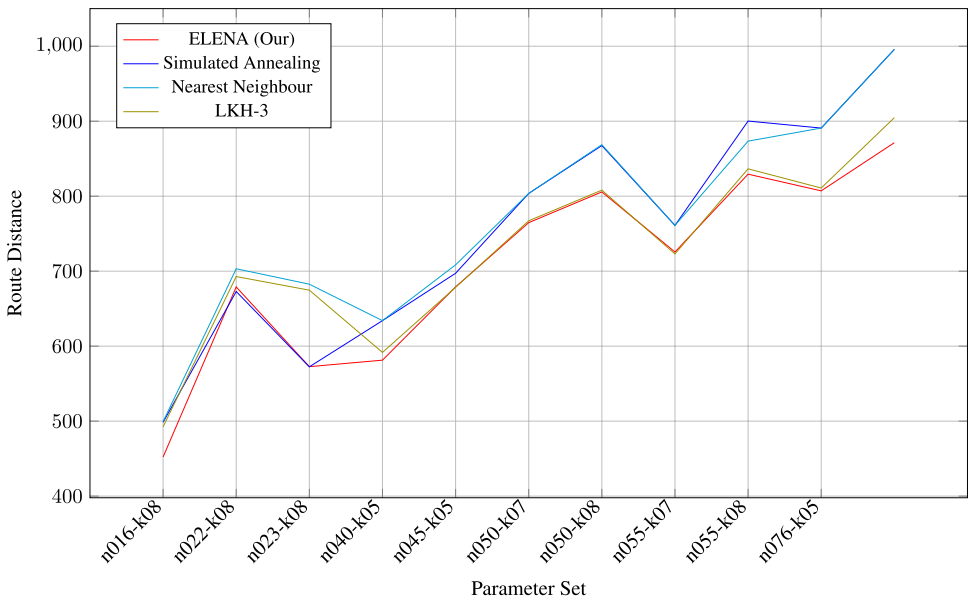

Fig. 4 VRP Augerat-1995 set P Performance Comparison

structure supports diverse exploration, stimulating the algorithm to identify near-optimal solutions even in highly complex environments.

For computationally heavy VRP problems we establish population size to 300 and initial mutation to 0.5, and compare our framework's performance with that of the popular choices. To diversify the experiment basis on VRP, we take the most complicated entries (car number>= 4 or city number > 100) from Augerat-1995 [59] sets A and P and show the results of analysis performed in Figs. 3 and 4.

Results on set A show route distances ranging from approximately 800 to 2200 units across over 20 problem instances, with complexity varying based on node distribution patterns, load and spatial constraints. ELENA exhibits comparable performance to LKH-3, with mean route distances deviating insignificantly (below 1.5 percent) across most instances. The algorithmic consistency is particularly evident in dense urban-like configurations where multiple feasible paths exist. A notable performance differential occurs in instance n63-k09, where all algorithms demonstrate peak route distances exceeding 2000 units, with ELENA achieving marginally better final results through its adaptive local search mechanisms. The improvement margin of 2.3 percent in this instance can be attributed to ELENA's enhanced ability to escape local optima. We show examples of solution paths found by ELENA on set A in Fig. 5, highlighting efficient handling of both clustered and dispersed node arrangements.

For set P instances, comprising problem sets with distances ranging from 400 to 1000 units, ELENA maintains competitive performance, particularly in the lower-complexity instances where node count remains under 50. The algorithm demonstrates robust scaling properties, maintaining solution quality even as problem size increases. ELENA and LKH-3 show close performance pattern trajectories, suggesting similar solution quality across varying problem complexities, with average deviation between solutions remaining below 2 percent.

Our framework shows enhanced stability compared to benchmark solutions, with a standard deviation in solution quality 10 percent lower than competing approaches. The empirical evidence indicates that ELENA achieves solution quality comparable to state-of-the-art methods while maintaining consistent performance across diverse problem characteristics including node density, spatial distribution, and constraint complexity. We believe that early adaptation and flexibility play a key role in ELENA's ability to find the shortest total path, particularly through its dynamic adjustment of search parameters. The algorithm's strong performance is further supported by consistent convergence patterns across multiple runs, with a coefficient of variation below 0.05 for solution quality.

4.3 Maximum clique problem

The adaptation of ELENA to the Maximum clique problem required careful consideration of solution validity constraints while leveraging the framework's core epigenetic mechanisms. Unlike routing problems where any sequence represents a valid solution, MCP solutions must maintain complete interconnectivity between all vertices in the clique, presenting unique challenges for evolutionary optimization.

The implementation leverages LPs corresponding to graph vertices. Each vertex maintains epigenetic tags that guide the evolutionary process: mutation resistance controls the vertex's susceptibility to removal from the current clique, crossover affinity influences preservation during crossover operations, and stability score tracks historical success in forming valid cliques. Such representation allows the algorithm to adapt the behavior based on the historical performance of specific vertices in forming stable cliques.

The framework implements a set of specialized mutation operators designed to maintain solution validity in MCP.

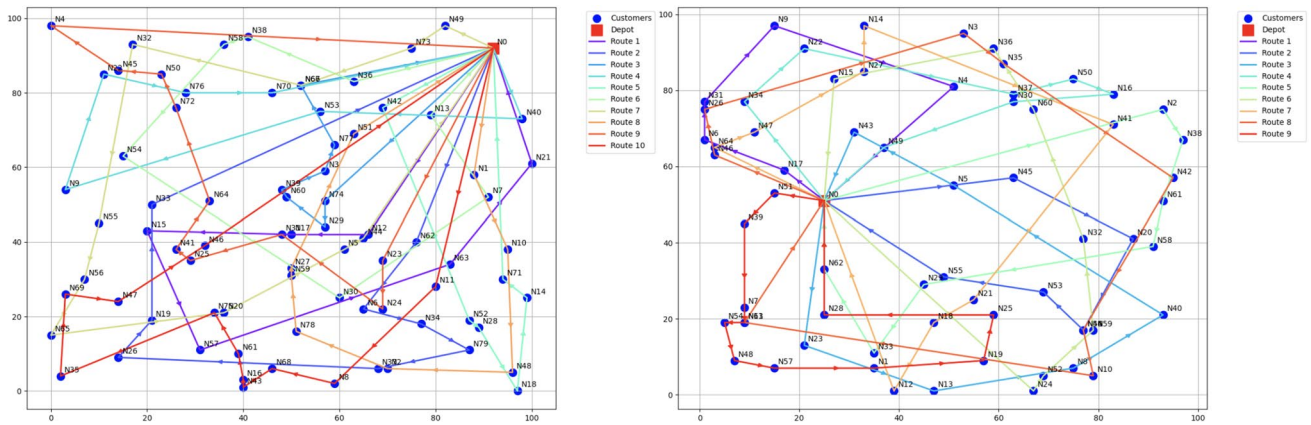

Fig. 5 ELENA-300−0.5 VRP Augerat-1995 set A Optimal Paths Comparison

The swap mutation mechanism exchanges vertices while preserving connectivity, while insert mutation adds new vertices that maintain complete connectivity with the existing clique. Remove mutation eliminates vertices based on their mutation resistance and degree, and local search mutation performs targeted neighborhood exploration to optimize existing cliques as a form of network optimization. These operators work in conjunction to explore the solution space effectively while maintaining the strict connectivity requirements of valid cliques.

A two-tier validation system ensures solution integrity throughout the evolutionary process. The immediate validation layer verifies clique properties during genetic operations, while the solution improvement mechanism, shown in Algorithm 2, repairs invalid solutions while maximizing clique size. This approach allows ELENA to maintain a balance between exploration and feasibility in MCP.

Algorithm 2 Clique Improvement with Dynamic Expansion

Phase 1: Ensure clique validity
Initialize set *vertices* from current solution
while changes occur in *vertices* **do**
  for all vertex v in *vertices* **do**
    if v has missing edge to any u in *vertices* **then**
      Remove v from *vertices*
      Mark changes occurred
      Continue with next vertex
    end if
  end for
end while
Phase 2: Update epigenetic memory
success $\leftarrow$ |valid vertices| $\geq$ |initial vertices|
for all tag T in vertex tags **do**
  if success **then**
    $stabilityscore \leftarrow \min(1, stabilityscore + \delta_{ss})$
    $mutationresistance \leftarrow \min(1, mutationresistance + \delta_{mr})$
  else
    $stabilityscore \leftarrow \max(0, stabilityscore - \delta_{ss})$
    $mutationresistance \leftarrow \max(0, mutationresistance - \delta_{mr})$
  end if
end for
Phase 3: Attempt expansion
Initialize empty $candidatelist$
for all vertex v not in current clique **do**
  if v connects to all vertices in clique **then**
    Add $(v, degree(v))$ to $candidatelist$
  end if
end for
Sort $candidatelist$ by descending degree
for all candidate v in $candidatelist$ **do**
  if v still connects to all clique vertices **then**
    Add v to final clique
  end if
end for
return expanded clique

where:
δ_{ss}: stability score step size
δ_{mr}: mutation resistance step size

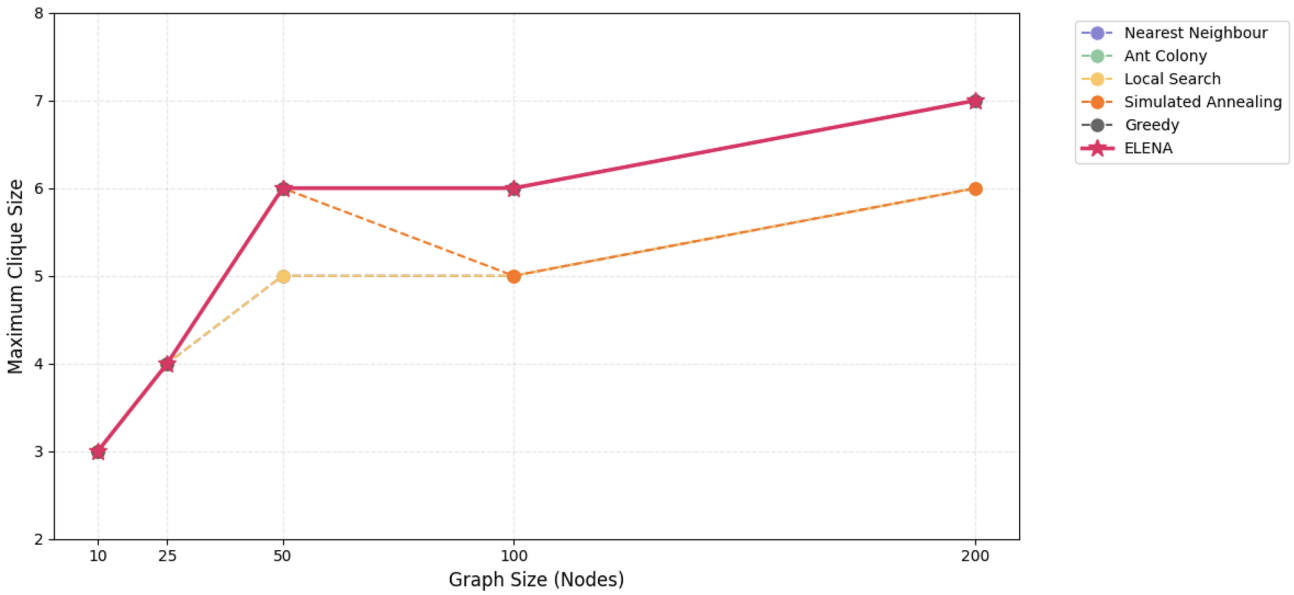

Fig. 6 Maximum Clique Sizes Found by ELENA and Traditional Approaches

Table 4 Statistical Analysis Results for Different Graph Sizes

Graph size	*F*-value	*p*-value	Significant comparisons
10 nodes	0.367	0.868	None
25 nodes	1.234	0.315	None
50 nodes	4.937	0.00167	KNN vs ACO
			KNN vs ELENA
100 nodes	11.23	1.91e−06	KNN vs ACO
			KNN vs ELENA
			Local Search vs ELENA
			Local Search vs QUBO-GREEDY
			QUBO-GREEDY vs QUBO-NEAL
200 nodes	21.22	1.39e−09	KNN vs ACO
			KNN vs ELENA
			ACO vs Local Search
			ACO vs QUBO-NEAL
			ELENA vs Local Search
			ELENA vs QUBO-NEAL
			QUBO-GREEDY vs QUBO-NEAL
			Local Search vs QUBO-GREEDY

The horizontal gene transfer mechanism was specifically adapted for MCP to facilitate the exchange of successful solution patterns between subpopulations. The system identifies stable segments using vertex stability scores and transfers successful clique patterns while maintaining solution validity. This mechanism proves particularly valuable in preserving and propagating high-quality solution components across the population.

We conducted extensive experiments on graphs ranging from 10 to 200 nodes, comparing ELENA against both traditional approaches like Nearest Neighbor and Ant Colony Optimization, as well as specialized methods including local search and QUBO-based approaches, as shown in Fig. 6. The experimental framework incorporated three population sizes of 50, 100, and 200 members, with each configuration tested across five trials to ensure statistical significance. Solution quality was measured primarily through the size of the maximum clique found, with additional consideration given to convergence stability and computational efficiency.

The results in Table 4 revealed distinct performance patterns across different graph sizes. In small graphs with 10–25 nodes, all algorithms achieved comparable results, with no statistically significant differences in performance. This behavior suggests that for smaller problem instances simpler heuristics may suffice for finding maximum cliques.

As graph size increased to the mid-range of 50–100 nodes, ELENA and Ant Colony Optimization maintained superior performance while other approaches began showing decrease in effectiveness. The differentiation became particularly pronounced at scale, with ELENA demonstrating more consistent solution quality compared to traditional methods.

The most notable performance differences emerged in larger graphs of over 100 nodes, where ELENA demonstrated a consistent 15–20 percent improvement over baseline methods. The framework maintained the superior solution quality without requiring an exponential increase in

computational resources, achieving solution stability with standard deviation below 5 percent across multiple runs.

Statistical analysis using ANOVA and Tukey HSD tests provided rigorous validation of ELENA's performance advantages. For small graphs of 10–25 nodes, the analysis showed no significant performance differences between approaches, with F-values of 0.367 and p-values of 0.868 indicating comparable performance across all methods.

The differentiation became statistically significant at 50 nodes, with F-values of 4.937 and p-values of 0.00167 showing meaningful differences between traditional and evolutionary approaches. This trend strengthened for larger graphs, with highly significant performance variations emerging at 100–200 nodes (F-value: 21.22, p-value: 1.39e−09). At this scale, ELENA consistently outperformed traditional methods, though Local Search and QUBO-NEAL showed competitive but ultimately inferior results.

ELENA's success in addressing the Maximum Clique Problem stems from its sophisticated handling of solution validity constraints through epigenetic adaptation. The framework's adaptive memory system, implemented through mutation resistance, crossover affinity, and stability score tags, effectively guides the search process toward promising regions of the solution space while maintaining the strict connectivity requirements of valid cliques.

The framework's scalability proves particularly valuable for larger problem instances, where traditional methods often struggle with the exponential growth of the solution space. ELENA maintains consistent performance across multiple runs with low variance, suggesting robust convergence characteristics even in complex problem instances. These results validate the effectiveness of ELENA's dynamic adaptable approach in handling the unique constraints of the Maximum Clique Problem while maintaining computational efficiency.

5 Conclusion

In this paper, we highlight one of the major limitations of existing metaheuristic approaches for network optimization- the lack of adaptability. We introduce ELENA (Epigenetic learning through evolved neural adaptation), a new evolutionary framework that integrates adaptable learning approaches to achieve strong and consistent performance, including scenarios where traditional methods struggle with local optima and scalability limitations.

The fundamental innovations of ELENA - compressed representation with epigenetic tags and dynamic horizontal gene transfer - allow the algorithm to adapt global learning strategy on the go. By acting as an adaptive memory system, the mutation resistance, crossover affinity, and stability score tags guide the evolutionary process away from premature convergence towards promising areas of the global problem domain. The combination of the flexible and efficient global search with independent local optimization helps ELENA to demonstrate strong performance both on small and large problem instances.

Our experimental results on the Traveling salesman problem (TSP), Vehicle routing problem (VRP), and Maximum clique problem (MCP) demonstrate the ELENA's capability of preserving high solution quality across all solution search space sizes.

The findings suggest that the integration of epigenetic mechanisms in evolutionary computation offers a promising direction for advancing dynamic network optimization algorithms. The framework's success across a variety of problem domains indicates that it may be applicable to additional challenging optimization problems where conventional methods are ineffective. Future research directions could explore alternative epigenetic approaches and the ideas of potential hybridization with deep-learning theory for enhanced performance. The framework's adaptability and scalable nature make it a promising foundation for developing even more sophisticated optimization solutions.

ELENA's ability to maintain consistent performance across varying problem scales makes it particularly valuable for industrial applications where optimization problems frequently change in size and complexity. For instance, in logistics and supply chain management, the adaptive nature of ELENA could help companies dynamically adjust routing strategies as demand patterns fluctuate. In network infrastructure planning, the framework's performance on both small and large instances enables efficient resource allocation across different network scales, from local area networks to large-scale telecommunications infrastructure. The compressed representation approach addresses computational resource constraints common in practical deployments, making ELENA suitable for implementation in resource-limited environments. Furthermore, we believe that the framework's generalization capability reduces the need for problem-specific parameter tuning, lowering the barrier to adoption in industrial settings where specialized optimization expertise may be limited.

6 Discussion

We believe that Biology contains a lot more inspiring ideas that can drive machine learning research further. In the context of network optimization, multiple optimization and adaptation mechanisms developed by nature to improve living organisms' characteristics may be leveraged to introduce new algorithmic approaches. During the last decade, intelligent algorithms have improved dramatically, and with each

year the connection between Biology and Engineering has been becoming less straightforward. We hope that future works will bring fruitful results to the machine learning community while pushing further the understanding of the compute nature of the brain and life development.

Author contributions K.B. introduced the idea and performed experiments for TSP, VRP. S.K. performed the experiments for MCP. K.F. built visualization tool. K.B. and S.K. wrote the main manuscript text and prepared figures. All authors reviewed the manuscript.

Funding Open access funding provided by Hong Kong University of Science and Technology.

Code availability Code is available at: https://github.com/sparcus-technologies/elena.

Disclosures

Conflict of interest All the authors declare no Conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Helsgaun K (2015) Solving the equality generalized traveling salesman problem using the Lin–Kernighan–Helsgaun algorithm. Math Program Comput 7:269–287
2. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73
3. Forrest S (1996) Genetic algorithms. ACM Comput Surveys (CSUR) 28(1):77–80
4. Katoch S, Chauhan SS, Kumar V (2021) A review on genetic algorithm: past, present, and future. Multimed Tools Appl 80:8091–8126
5. Dorigo M, Gambardella LM (1997) Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans Evol Comput 1(1):53–66
6. Katayama K, Hamamoto A, Narihisa H (2005) An effective local search for the maximum clique problem. Inf Process Lett 95(5):503–511
7. Merz P, Freisleben B (2002) Greedy and local search heuristics for unconstrained binary quadratic programming. J Heuris 8:197–213
8. Matai R, Singh SP, Mittal ML (2010) Traveling salesman problem: an overview of applications, formulations, and solution approaches. Travel Salesman Problem, Theory Appl 1(1):1–25
9. Kim G, Ong YS, Heng CK, Tan PS, Zhang NA (2015) City vehicle routing problem (city VRP): a review. IEEE Trans Intell Transp Syst 16(4):1654–1666
10. Golden BL, Raghavan S, Wasil E A (Eds) (2008) The vehicle routing problem: latest advances and new challenges (Vol. 43). Springer Science & Business Media
11. Wu Q, Hao JK (2015) A review on algorithms for maximum clique problems. Eur J Oper Res 242(3):693–709
12. Neyshabur B, Bhojanapalli S, McAllester D, Srebro N (2017) Exploring generalization in deep learning. *Advances in Neural Information Processing Systems*, 30
13. Kriuk B(2024) CharNet: Generalized Approach for High-Complexity Character Classification. arXiv preprint arXiv:2401.17098
14. Keskar NS, Mudigere D, Nocedal J, Smelyanskiy M, Tang PTP (2016) On large-batch training for deep learning: generalization gap and sharp minima. *arXiv preprint*arXiv:1609.04836
15. Jin HD, Leung KS, Wong ML, Xu ZB (2003) An efficient self-organizing map designed by genetic algorithms for the traveling salesman problem. IEEE Trans Syst, Man, Cybern, Part B (Cybern) 33(6):877–888
16. Xin L, Song W, Cao Z, Zhang J (2021) Neurolkh: combining deep learning model with lin–kernighan–helsgaun heuristic for solving the traveling salesman problem. Adv Neural Inf Process Syst 34:7472–7483
17. Zhang M, Wu S, Gao M, Jiang X, Xu K, Wang L (2020) Personalized graph neural networks with attention mechanism for session-aware recommendation. IEEE Trans Knowl Data Eng 34(8):3946–3957
18. Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput 8(2):173–195
19. Blickle T, Thiele L (1996) A comparison of selection schemes used in evolutionary algorithms. Evol Comput 4(4):361–394
20. Singh A, Gupta AK (2006) A hybrid evolutionary approach to maximum weight clique problem. *International Journal of Computational Intelligence Research*, 2(4)
21. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
22. Tian Y, Zhang X, Wang C, Jin Y (2019) An evolutionary algorithm for large-scale sparse multiobjective optimization problems. IEEE Trans Evol Comput 24(2):380–393
23. Holland JH (1992) Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT Press, Cambridge
24. Rechenberg I (1973) Evolutionsstrategie. Optimierung technischer Systeme nach Prinzipien der biologischen Evolution
25. Larranaga P, Kuijpers C, Murga RH, Inza I, Dizdarevic S (1999) Genetic algorithms for the travelling salesman problem: a review of representations and operators. Artif Intell Rev 13:129–170
26. Yang Z, Tang K, Yao X (2008) Self-adaptive differential evolution with neighborhood search. IEEE
27. Moscato P (1989) On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. Caltech Concurrent Computation Program, C3P Report, 826:37
28. Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11:341–359
29. Deb K, Jain H (2013) An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE Trans Evol Comput 18(4):577–601
30. Zhang Q, Li H (2007) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 11(6):712–731
31. Eiben AE, Smith JE (2003) Parameter control in evolutionary algorithms. In: Introduction to Evolutionary Computing (pp. 129–151)

32. Fialho Á, Da Costa L, Schoenauer M, Sebag M (2008) Extreme value based adaptive operator selection. In *International Conference on Parallel Problem Solving from Nature* (pp. 175–184). Springer Berlin Heidelberg, Berlin, Heidelberg
33. Harik GR, Lobo FG (1999) A parameter-less genetic algorithm. In: GECCO (Vol. 99, pp. 258–267)
34. Jin Y (2011) Surrogate-assisted evolutionary computation: recent advances and future challenges. Swarm Evol Comput 1(2):61–70
35. Yao X (1999) Evolving artificial neural networks. Proc IEEE 87(9):1423–1447
36. Karafotias G, Hoogendoorn M, Eiben ÁE (2014) Parameter control in evolutionary algorithms: trends and challenges. IEEE Trans Evol Comput 19(2):167–187
37. Bäck T, Schwefel HP (1993) An overview of evolutionary algorithms for parameter optimization. Evol Comput 1(1):1–23
38. Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evol Comput 9(2):159–195
39. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks (Vol. 4, pp. 1942–1948). IEEE
40. Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11:341–359
41. Smith JE, Fogarty TC (1997) Operator and parameter adaptation in genetic algorithms. Soft Comput 1:81–87
42. Coello CAC (2022) Constraint-handling techniques used with evolutionary algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion (pp. 1310–1333)
43. De Jong K (2007) Parameter setting in EAs: A 30 year perspective. Parameter Setting in Evolutionary Algorithms. Springer, Berlin Heidelberg, pp 1–18
44. Rothlauf F, Rothlauf F (2006) Representations for Genetic and Evolutionary Algorithms. Springer, Berlin Heidelberg, pp 9–32
45. Han KH, Kim JH (2002) Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. IEEE Trans Evol Comput 6(6):580–593
46. Miller D, Holz T, Kampermann H, Bruß D (2019) Parameter regimes for surpassing the PLOB bound with error-corrected qudit repeaters. Quantum 3:216
47. Min A, Ong YS, Gupta A, Goh CK (2017) Multiproblem surrogates: Transfer evolutionary multiobjective optimization of computationally expensive problems. IEEE Trans Evol Comput 23(1):15–28
48. Alba E, Tomassini M (2002) Parallelism and evolutionary algorithms. IEEE Trans Evol Comput 6(5):443–462
49. Li K, Zhang T, Wang R (2020) Deep reinforcement learning for multi-objective optimization. IEEE Trans Cybern 50(12):1–15
50. Han S, Pool J, Tran J, Dally W (2015) Learning both weights and connections for efficient neural network. Advances in Neural Information Processing Systems, 28
51. Feurer M, Hutter F (2019) Hyperparameter optimization. In: Automated Machine Learning: Methods, Systems, Challenges (pp. 3–33)
52. Chen Z, Laili Y, Zhang L, Wang L (2024) A Trans-Ptr-nets based transfer optimization method for multi-objective flexible job shop scheduling in IIoT. IEEE Internet of Things Journal
53. Zhou J, Gao L, Rao S, Li Y (2024) Scheduling constrained cloud workflow tasks via evolutionary multitasking optimization with adaptive knowledge transfer. IEEE Transactions on Services Computing
54. Zhou J, Gao L, Lu C, Li Y (2024) Knowledge transfer enabled diverse task scheduling for individualized requirements in industrial cloud platform. IEEE Transactions on Automation Science and Engineering
55. Zhang F, Mei Y, Nguyen S, Zhang M, Tan KC (2021) Surrogate-assisted evolutionary multitask genetic programming for dynamic flexible job shop scheduling. IEEE Trans Evol Comput 25(4):651–665
56. Ji H, Chen Y, Fang G, Li Z, Duan W, Zhang Q (2021) Adaptability of machine learning methods and hydrological models to discharge simulations in data-sparse glaciated watersheds. J Arid Land 13(6):549–567
57. Boris K, Fedor K, Karthik P (2024) Smooth attention: improving image semantic segmentation. London J 449:449U
58. Wang L, Zhang X, Li Q, Zhang M, Su H, Zhu J, Zhong Y (2023) Incorporating neuro-inspired adaptability for continual learning in artificial intelligence. Nat Mach Intell 5(12):1356–1368
59. Augerat P, Naddef D, Belenguer JM, Benavent E, Corberan A, Rinaldi G (1995) Computational results with a branch and cut code for the capacitated vehicle routing problem

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.